Modelling Sustainable Systems and Semantic Web

RDF Basics

Lecture in the Module 10-202-2312
for Master Computer Science

Prof. Dr. Hans-Gert Gräbe
http://www.informatik.uni-leipzig.de/~graebe

June 2022

RDF Basics. Descriptions and Interpretations

Information as interpreted data?

- Measured values as data?

Language is full of implicit assumptions. An example:

- On November 8th at the station Leipzig Airport at 5 p.m. was measured a temperature of 16°C.
- On November 8th at the station Leipzig Airport at 5 p.m. was measured a temperature of 16 °C.
- Things and their names.

RDF Basics. Example

```
@prefix lv:  <http://od.fmi.uni-leipzig.de/lv/> .
@prefix od:  <http://od.fmi.uni-leipzig.de/model/>.
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#>.
@prefix odr:  <http://od.fmi.uni-leipzig.de/rooms/>.
@prefix odp:  <http://od.fmi.uni-leipzig.de/personal/>.

lv:BIS.SemanticWeb.1
    rdf:type od:English , od:LV , od:Vorlesung ;
    rdfs:label "Modelling Substainable Systems ...  " ;
    od:beginsAt "9:15" ;
    od:dayOfWeek "donnerstags" ;
    od:endsAt "10:45" ;
    od:locatedAt odr:online ;
    od:servedBy odp:Graebe_HansGert ;
    od:hasCType "synchron" .
```

- Identifiers and literals. Namespaces.

RDF Basics. Sentences

Resolution in three-word sentences

```
Subject Predicate Object .
```

```
lv:BIS.SemanticWeb.1 rdf:type od:Vorlesung .
lv:BIS.SemanticWeb.1 rdfs:label "Modelling ...  " .
lv:BIS.SemanticWeb.1 od:beginsAt "9:15" .
lv:BIS.SemanticWeb.1 od:dayOfWeek "donnerstags" .
lv:BIS.SemanticWeb.1 od:endsAt "10:45" .
lv:BIS.SemanticWeb.1 od:locatedAt odr:online .
lv:BIS.SemanticWeb.1 od:servedBy odp:Graebe_HansGert .
lv:BIS.SemanticWeb.1 od:hasCType "synchron" .
```

Another example: `http://leipzig-data.de/info/`
(Tab "MINT-Kataloge")

Industry 4.0 and the Internet of Things

- Shortcut speaking. There are no things on the Internet, only *representations* of things, just like representations of people (digital identities).
- Descriptions as relational complexes between representations of real things or even just complexes of meaning.
- These things and complexes of meaning must also be assigned "Digital Identities" as textual representations to be able to formulate sentences about them in the Digital Universe.

RDF Basics. Conceptual Ingredients

- UTF-8 as a **uniform character base** for URIs and literals.

  Best practice: URIs only made up of ASCII characters, no diacritics, special characters or similar Unicode.
- URI as "digital identities" of resources, *point* to resources.

  Like people's digital identities, these are **textual representations of "things"** in the text fragments circulating in the internet.
- For computers, URIs are just strings, for people it is helpful if the URI already provides a suggestion about its semantics.

  Best practice: "Speaking names" as URIs

RDF Basics. Best Practices

- RDF – Resource Description Framework
- Concept for writing down stories about "the world" as **sets** of three-word sentences

  <subject> <predicate> <object>.
- Subject and predicate must be URIs. The object can be a URI or a literal (type `rdfs:Literal`). Literals can carry type and language markings.
- There are different notations for the same set of RDF sentences (Turtle, rdf/xml, json, ntriples) and tools to convert these notations.

  Redland RDF libraries `http://librdf.org/`
- Pattern search as a powerful concept for analyzing such sets of sentences. SPARQL as query language.

SPARQL cannot be discussed here.

RDF Basics. Different Notations

- **Turtle notation** – collects together all sentences about the same subject. Such a set of predicate-object pairs can be interpreted as a set of key-value pairs that describes this subject.
  - But: Here a key can have *several* values!
  - It is a particularly popular human readable notation.
  - It is a subject-centered point of view, which well serves the specific point of view of "MY World" – as discussed earlier.
  - Computers prefer to work with sets of triples.
- If the subjects and objects are interpreted as nodes and predicates as edges then a set of RDF sentences describes an **RDF graph** (and vice versa).

  A picture is often a better explanation than thousand words.

RDF – Sentences and Pattern

Sentences are arranged following patterns:

1) **Turtle:** Collect all sentences with the same subject. Interpretation of properties of an individual subject as key-value pairs.
   - Key and value = attribute and attribute value
2) Collect **all sentences with the same predicate**

   ```
   A od:beginsAt B
   ```

   - `od:beginsAt` is not only a URI (*syntax*), but also a predicate with two parameters (A and B) and a certain *semantics* that is present in all sentences with this predicate as its *instantiations*.
3) Other patterns are possible, SPARQL as the general standard query language for pattern search in RDF sets of sentences.

RDF – Descriptions of Descriptions

- **Self-similarity of the concept:** Also descriptions of descriptions can be formulated as RDF phrases. In particular, one can use RDF to describe RDF.
  - A URI that appears as a predicate in a sentence can appear as subject or object in other sentences.
    Example:

```
od:beginsAt rdfs:domain od:LV .
od:beginsAt rdfs:range rdfs:Literal .
```

- This means that also *terms and concepts* can be described using RDF. → **Universals**
  - What are universals? Ideas from Plato's *heaven of ideas* or *institutionalized conventions*, i.e. "shortcuts" of speaking?

RDF – Basic Limitations

- Set semantics, the order of the sentences does not matter.
  - This is different in other approaches, such as the XML-based TEI (Text Encoding Initiative) which plays a central role in Digital Humanities.
- Problem of contextualization. In which spatio-temporal context the sentence has to be interpreted? There are several approaches here:
  - Extend triples to quadruples with a fourth component containing the URI to the provenance (description).
  - If the sentence contains an instantiation of a predicate, the context often can be inferred from the set of instantiations of that predicate.
  - Often the context results more generally from the *namespace* of the predicate and thus stands as an (explicit or implicit) *model for a whole class of terms*. But this shifts the problem only to the description of the model and thus an abstraction level upwards.

RDF – Summary of the Central Concepts

- *Central idea:* Save textual descriptions in a uniform way as triples and use standard concepts and tools for the management of this data.
- *Resources:* URI, HTTP access
  - URI = Unique Resource Identifier
  - This can be used to access a worldwide distributed database in a uniform manner via a common protocol.
- *Resource Descriptions:* Return on a HTTP request a useful piece of information in RDF format that can be combined with others such information units.
- Operate an *RDF Triple Stores* and a *SPARQL endpoint*, e.g. the RDF Store behind `http://leipzig-data.de/info/index.php` and the Endpoint `http://leipzig-data.de:8890/sparql` as part of a worldwide distributed Data storage infrastructure.
- SPARQL as language for (distributed) queries.